Sys Manager Code Challenge
Raw

challenge.md

STL File Analysis

Challenge

For this challenge we'd like you to implement a parser that can do some basic analysis of a 3D model. Daily at Sys Manager, we work with 3D part files represented as STL files. STL (stereolithography) files describe the surface geometry of a three-dimensional object in terms of a mesh of triangles. They are a simple representation, without any concepts of color, texture, or other common CAD model attributes. STL files have both ASCII and binary representations, but for this challenge we'll only work with the ASCII version.

Your code should be able to read in an STL file and report on the number of triangles contained in the file and the surface area of the model.

Instructions

You may develop your solution in any language that you like. Your code should output these metrics:

- number of triangles in the file
- surface area of the 3D part

Along with your solution please provide a README that includes the following:

- instructions for running your code
- an explanation of the design decisions you made
- potential performance improvements you might make so your solution could handle a model with millions of triangles

Implementation Notes

Surface area is calculated as the sum of the areas of all the triangles. Feel free to Google for the formula.

For verifying the accuracy of your code, you can use the simple STL example provided below, as well as the Moon.stl file provided with this link.

In order to reduce the complexity of the challenge, you can safely assume that the input STL files are all properly formed and none of the triangles are overlapping.

Using Dependencies

You may use 3rd party libraries to assist you in completing this challenge, however we would like for the bulk of the challenge to be your own code. For example, using a library that is designed to parse STLs is off limits, but if you wanted to use a library that handles file reading, feel free to. Because this criteria is subjective, please feel free to reach out to us with any questions.

What We're Looking For

While this challenge is about working with 3D models, we're not trying to test your math knowledge or ability to think about 3D space. We are primarily judging your ability to work with data structures with readable and extensible code.

Note: If you continue on the in the interview process after this stage, we'll revisit your code challenge in a live Code Extension interview where we'll ask you to extend your code with a new feature.

More on the STL Format

The ASCII representation of an STL describes a triangle mesh in plaintext. If you're curious, you can read more about STLs here.

The file is bookended by `solid` and `endsolid`, and each `facet` block describes a triangle with 3 vertexes. Each `vertex` contains the x, y, z coordinates of that point in the triangle.

```
solid name
  facet normal ni nj nk
      outer loop
          vertex v1x v1y v1z
          vertex v2x v2y v2z
          vertex v3x v3y v3z
      endloop
  endfacet
endsolid name
```

For the purposes of this challenge, you can ignore the `normal` definition at the start of the facet block. `facet normal` describes the direction a triangle is pointing. In this challenge assume all triangles are pointing in an accurate direction.

Example File and Expected Output

Simple STL example:

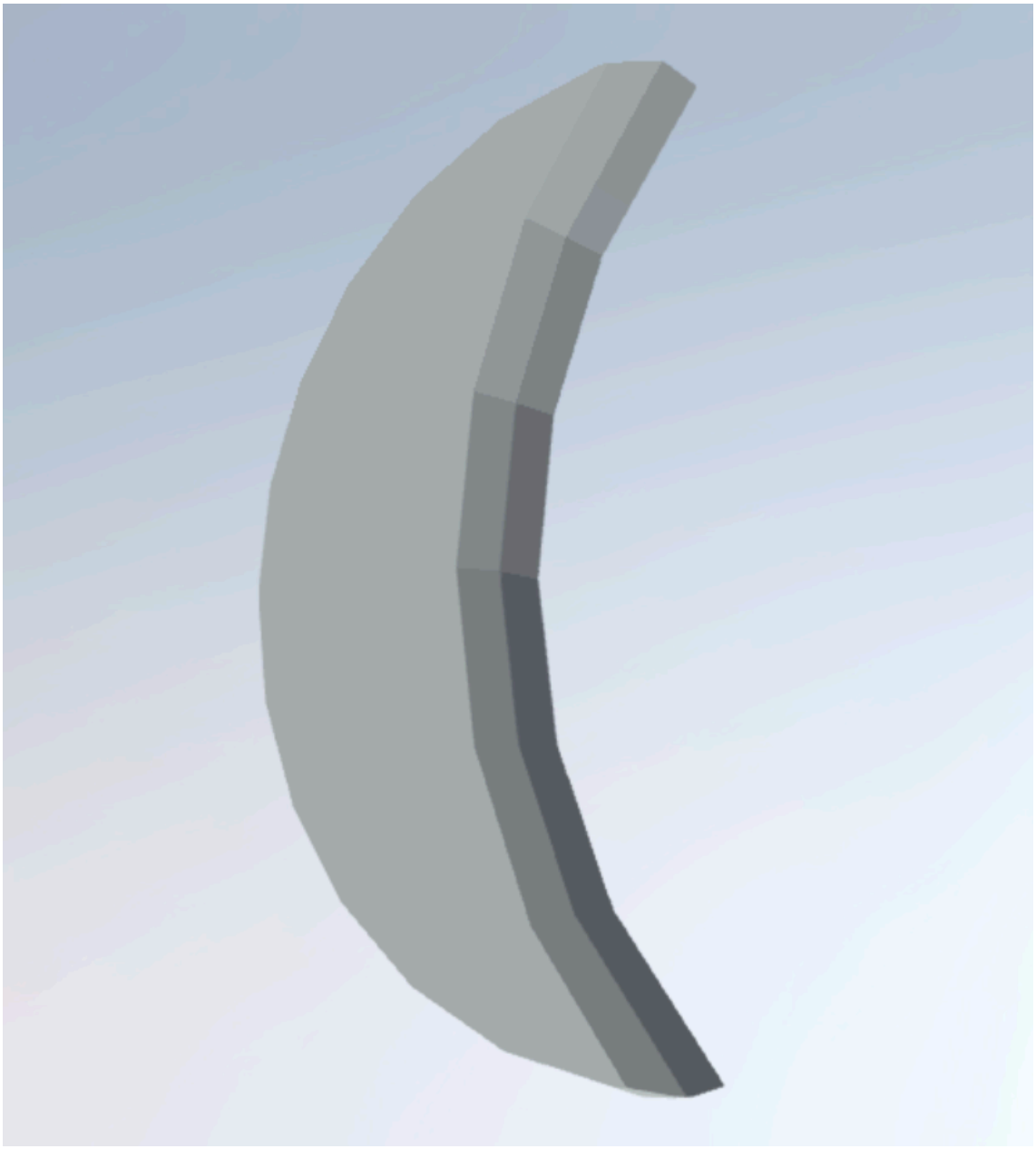
```
solid simplePart
  facet normal 0 0 0
      outer loop
          vertex 0 0 0
          vertex 1 0 0
          vertex 1 1 1
      endloop
  endfacet
  facet normal 0 0 0
      outer loop
          vertex 0 0 0
```

```
          vertex 0 1 1
          vertex 1 1 1
      endloop
  endfacet
endsolid simplePart
Number of Triangles: 2
Surface Area: 1.4142
```

More Complex Example

For a more complex file, use Moon.stl.

Here's what Moon looks like:



Expected output for Moon:

Number of Triangles: 116
Surface Area: 7.7726

Sign up for free